**Parcial 2**

Carnet 16-10109 Jonathan Bautista

Pregunta 1:

Lenguaje de Programación Escogido Boo:

Boo es un lenguaje de programación orientado a objetos, de tipos estáticos para la Common Language Infrastructure con una sintaxis inspirada en Python y un énfasis en la extensibilidad del lenguaje y su compilador. Sus características incluyen la inferencia de tipos, los generadores, multimétodos, duck typing opcional, macros, clausuras, currificación y funciones de primera clase.

Boo es un lenguaje basado en CLR, lo que significa que cuando usamos Boo, nos beneficiamos de la rica biblioteca que viene con .NET, optimizaciones JIT (compilación justo a tiempo) y muchas buenas herramientas. Incluso sin extender el compilador, Boo tiene lo siguiente para ofrecer:

En el manifiesto, los objetivos que marca el autor son, en relación a la sintaxis, que tiene una sintaxis tipo Python, por lo que requiere, al igual que Python, de la identación de forma adecuada para delimitar los bloques. La declaración automática de variables... que no es una gran característica para los lenguajes scripting... pero cuando se viene de C++, C# o Java, puede ser toda una novedad, al igual que la inferencia de tipos.

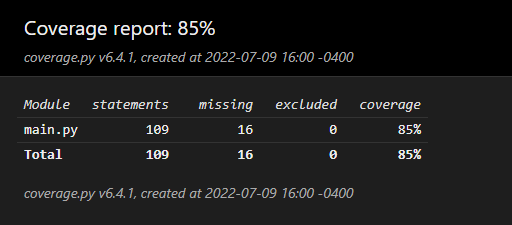
Al igual que en cualquier lenguaje script, y a diferencia de C# y Java, las clases son opcionales. Se puede escribir código directamente para ser ejecutado o funciones, sin necesidad de crear una clase para ello.

Se pueden emplear clases como argumentos (lo que se conoce como funciones de primera clase)

b) Codigo disponible en : <https://github.com/JKOMI2303/Lenguaje_Programacion-1/tree/main/parcial%202/pregunta_1>

Pregunta 2:

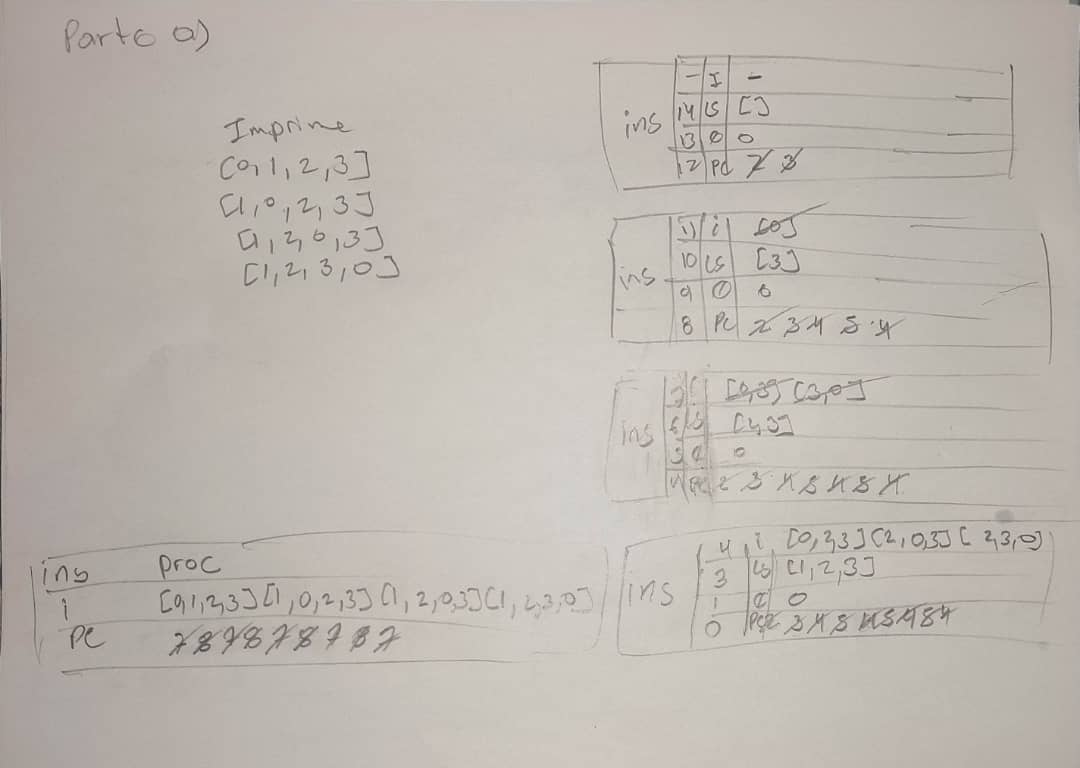
Se realizaron las pruebas unitarias y de cobertura y arroja una cobertura del 85%.



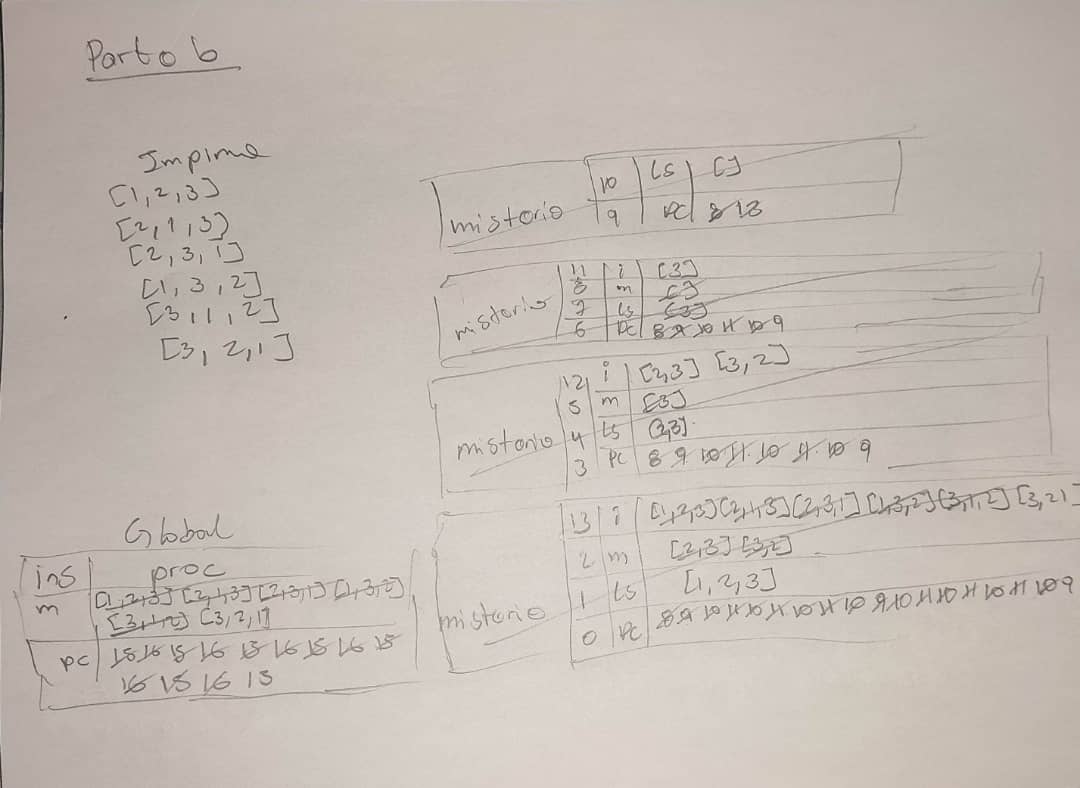
Codigo disponible en: <https://github.com/JKOMI2303/Lenguaje_Programacion-1/tree/main/parcial%202/pregunta_2>

Pregunta 3:

a)



b)



El iterador misterio usa el iterador ins, para obtener todas las permutaciones posibles en los elementos de un arreglo. en este caso es aprovechado el uso del iterador ins de manera que no se repita alguna permutación, ins es llamado de esta manera: for i in ins(1, [2,3]) y también for i in ins(1, [3,2]), así para la primera retorna [1,2,3], [2,1,3], [2,3,1] y para la segunda retorna [1.3.2], [3,1,2], [3,2,1] siendo estas seis todas las permutaciones de [1,2,3].

Codigo de iterador bienParentizado en :

<https://github.com/JKOMI2303/Lenguaje_Programacion-1/tree/main/parcial%202/pregunta_3>

En este código se utilizan los iteradores misterio e ins para hacer uso de sus propiedades de permutaciones luego se filtra en misterio para quitar las permutaciones repetidas y finalmente en bienparentizado para ver si es una expresión correcta.

Pregunta 4:

X=1, Y=0 , Z=9

alpha= ((X+Y)mod5)+3) beta=((Y+Z)mod5)+3)

alpha=(1mod5)+3 beta=(9mod5)+3

alpha=(1+3)=4 beta=4+3=7

Funcion\_alpha,beta(n):

F\_4,7(n)= n si 0<=n<4\*7

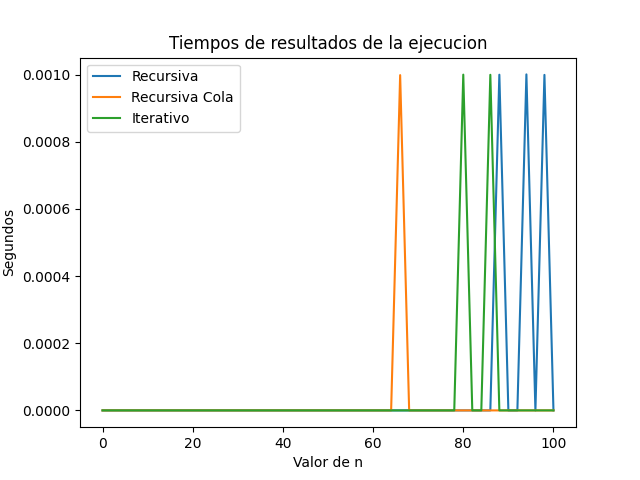
= F\_4,7(n-7\*1)+F\_4,7(n-7\*2)+F\_4,7(n-7\*3)+F\_4,7(n-7\*4) si n>=4\*7

F\_4,7(n)= n si 0<=n<28

= F\_4,7(n-7)+F\_4,7(n-14)+F\_4,7(n-21)+F\_4,7(n-28) si n>=28

Comparación entre las funciones:

Para n=100 con salto=2



Resultados de la ejecucion:

-----------------------------------------------------------------------------------

n || Funcion Recursiva || Funcion Recursiva Cola || Funcion Iterativo

-----------------------------------------------------------------------------------

0 || 0 || 0 || 0

2 || 2 || 2 || 2

4 || 4 || 4 || 4

6 || 6 || 6 || 6

8 || 8 || 8 || 8

10 || 10 || 10 || 10

12 || 12 || 12 || 12

14 || 14 || 14 || 14

16 || 16 || 16 || 16

18 || 18 || 18 || 18

20 || 20 || 20 || 20

22 || 22 || 22 || 22

24 || 24 || 24 || 24

26 || 26 || 26 || 26

28 || 42 || 42 || 42

30 || 50 || 50 || 50

32 || 58 || 58 || 58

34 || 66 || 66 || 66

36 || 91 || 91 || 91

38 || 105 || 105 || 105

40 || 119 || 119 || 119

42 || 161 || 161 || 161

44 || 187 || 187 || 187

46 || 213 || 213 || 213

48 || 239 || 239 || 239

50 || 333 || 333 || 333

52 || 383 || 383 || 383

54 || 433 || 433 || 433

56 || 595 || 595 || 595

58 || 693 || 693 || 693

60 || 791 || 791 || 791

62 || 889 || 889 || 889

64 || 1242 || 1242 || 1242

66 || 1430 || 1430 || 1430

68 || 1618 || 1618 || 1618

70 || 2212 || 2212 || 2212

72 || 2574 || 2574 || 2574

74 || 2936 || 2936 || 2936

76 || 3298 || 3298 || 3298

78 || 4612 || 4612 || 4612

80 || 5310 || 5310 || 5310

82 || 6008 || 6008 || 6008

84 || 8218 || 8218 || 8218

86 || 9564 || 9564 || 9564

88 || 10910 || 10910 || 10910

90 || 12256 || 12256 || 12256

92 || 17138 || 17138 || 17138

94 || 19732 || 19732 || 19732

96 || 22326 || 22326 || 22326

98 || 30534 || 30534 || 30534

100 || 35534 || 35534 || 35534

Tiempos de ejecucion en segundos:

-----------------------------------------------------------------------------------

n || Funcion Recursiva || Funcion Recursiva Cola ||Funcion Iterativa

-----------------------------------------------------------------------------------

0 || 0.00000 || 0.00000 || 0.00000

2 || 0.00000 || 0.00000 || 0.00000

4 || 0.00000 || 0.00000 || 0.00000

6 || 0.00000 || 0.00000 || 0.00000

8 || 0.00000 || 0.00000 || 0.00000

10 || 0.00000 || 0.00000 || 0.00000

12 || 0.00000 || 0.00000 || 0.00000

14 || 0.00000 || 0.00000 || 0.00000

16 || 0.00000 || 0.00000 || 0.00000

18 || 0.00000 || 0.00000 || 0.00000

20 || 0.00000 || 0.00000 || 0.00000

22 || 0.00000 || 0.00000 || 0.00000

24 || 0.00000 || 0.00000 || 0.00000

26 || 0.00000 || 0.00000 || 0.00000

28 || 0.00100 || 0.00000 || 0.00000

30 || 0.00000 || 0.00000 || 0.00000

32 || 0.00000 || 0.00000 || 0.00000

34 || 0.00000 || 0.00000 || 0.00000

36 || 0.00000 || 0.00000 || 0.00000

38 || 0.00000 || 0.00000 || 0.00000

40 || 0.00000 || 0.00000 || 0.00000

42 || 0.00000 || 0.00000 || 0.00000

44 || 0.00000 || 0.00000 || 0.00000

46 || 0.00000 || 0.00000 || 0.00000

48 || 0.00000 || 0.00000 || 0.00000

50 || 0.00000 || 0.00000 || 0.00000

52 || 0.00000 || 0.00000 || 0.00000

54 || 0.00000 || 0.00000 || 0.00000

56 || 0.00000 || 0.00000 || 0.00000

58 || 0.00000 || 0.00000 || 0.00000

60 || 0.00000 || 0.00000 || 0.00000

62 || 0.00000 || 0.00000 || 0.00000

64 || 0.00000 || 0.00000 || 0.00000

66 || 0.00000 || 0.00000 || 0.00000

68 || 0.00000 || 0.00000 || 0.00000

70 || 0.00000 || 0.00100 || 0.00000

72 || 0.00000 || 0.00000 || 0.00000

74 || 0.00000 || 0.00000 || 0.00000

76 || 0.00000 || 0.00000 || 0.00000

78 || 0.00000 || 0.00000 || 0.00000

80 || 0.00000 || 0.00000 || 0.00000

82 || 0.00000 || 0.00000 || 0.00100

84 || 0.00000 || 0.00000 || 0.00000

86 || 0.00000 || 0.00000 || 0.00000

88 || 0.00000 || 0.00000 || 0.00000

90 || 0.00200 || 0.00000 || 0.00000

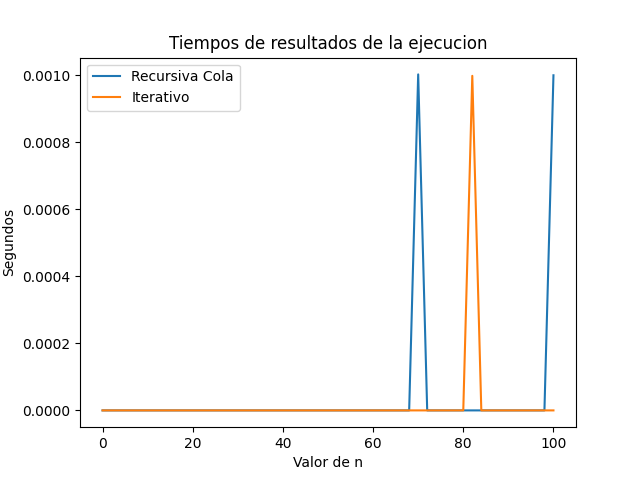
92 || 0.00000 || 0.00000 || 0.00000

94 || 0.00100 || 0.00000 || 0.00000

96 || 0.00000 || 0.00000 || 0.00000

98 || 0.00100 || 0.00000 || 0.00000

100 || 0.00000 || 0.00100 || 0.00000



Comparacion de Recursiva de cola con la Iterativa

Resultados de la ejecucion:

--------------------------------------------------------------------------------

n || Recursiva Cola || Iterativa

--------------------------------------------------------------------------------

0 || 0 || 0

2 || 2 || 2

4 || 4 || 4

6 || 6 || 6

8 || 8 || 8

10 || 10 || 10

12 || 12 || 12

14 || 14 || 14

16 || 16 || 16

18 || 18 || 18

20 || 20 || 20

22 || 22 || 22

24 || 24 || 24

26 || 26 || 26

28 || 42 || 42

30 || 50 || 50

32 || 58 || 58

34 || 66 || 66

36 || 91 || 91

38 || 105 || 105

40 || 119 || 119

42 || 161 || 161

44 || 187 || 187

46 || 213 || 213

48 || 239 || 239

50 || 333 || 333

52 || 383 || 383

54 || 433 || 433

56 || 595 || 595

58 || 693 || 693

60 || 791 || 791

62 || 889 || 889

64 || 1242 || 1242

66 || 1430 || 1430

68 || 1618 || 1618

70 || 2212 || 2212

72 || 2574 || 2574

74 || 2936 || 2936

76 || 3298 || 3298

78 || 4612 || 4612

80 || 5310 || 5310

82 || 6008 || 6008

84 || 8218 || 8218

86 || 9564 || 9564

88 || 10910 || 10910

90 || 12256 || 12256

92 || 17138 || 17138

94 || 19732 || 19732

96 || 22326 || 22326

98 || 30534 || 30534

100 || 35534 || 35534

Tiempos de ejecucion en segundos:

--------------------------------------------------------------------------------

n || Recursiva Cola || Iterativa

--------------------------------------------------------------------------------

0 || 0.00000 || 0.00000

2 || 0.00000 || 0.00000

4 || 0.00000 || 0.00000

6 || 0.00000 || 0.00000

8 || 0.00000 || 0.00000

10 || 0.00000 || 0.00000

12 || 0.00000 || 0.00000

14 || 0.00000 || 0.00000

16 || 0.00000 || 0.00000

18 || 0.00000 || 0.00000

20 || 0.00000 || 0.00000

22 || 0.00000 || 0.00000

24 || 0.00000 || 0.00000

26 || 0.00000 || 0.00000

28 || 0.00000 || 0.00000

30 || 0.00000 || 0.00000

32 || 0.00000 || 0.00000

34 || 0.00000 || 0.00000

36 || 0.00000 || 0.00000

38 || 0.00000 || 0.00000

40 || 0.00000 || 0.00000

42 || 0.00000 || 0.00000

44 || 0.00000 || 0.00000

46 || 0.00100 || 0.00000

48 || 0.00000 || 0.00000

50 || 0.00000 || 0.00000

52 || 0.00000 || 0.00000

54 || 0.00000 || 0.00000

56 || 0.00000 || 0.00000

58 || 0.00000 || 0.00000

60 || 0.00000 || 0.00000

62 || 0.00000 || 0.00000

64 || 0.00000 || 0.00000

66 || 0.00000 || 0.00000

68 || 0.00000 || 0.00100

70 || 0.00000 || 0.00000

72 || 0.00000 || 0.00000

74 || 0.00000 || 0.00000

76 || 0.00000 || 0.00000

78 || 0.00000 || 0.00000

80 || 0.00000 || 0.00000

82 || 0.00000 || 0.00100

84 || 0.00000 || 0.00000

86 || 0.00000 || 0.00000

88 || 0.00000 || 0.00000

90 || 0.00000 || 0.00000

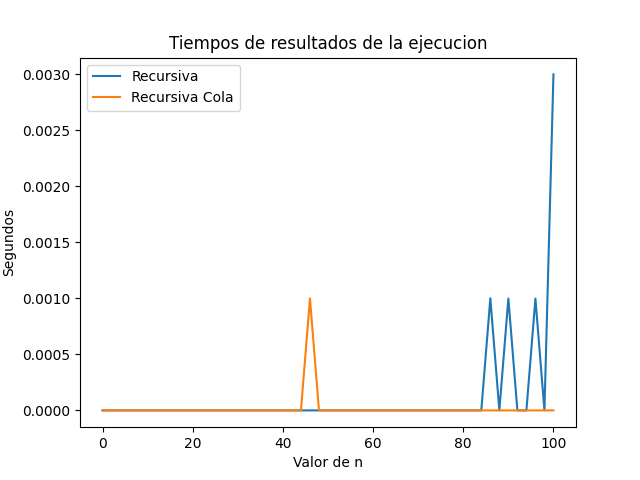
92 || 0.00000 || 0.00000

94 || 0.00000 || 0.00000

96 || 0.00000 || 0.00000

98 || 0.00000 || 0.00000

100 || 0.00000 || 0.00000



Comparacion de recursiva con recursiva de cola

Resultados de la ejecucion:

--------------------------------------------------------------------------------

n || Recursiva || Recursiva de cola

--------------------------------------------------------------------------------

0 || 0 || 0

2 || 2 || 2

4 || 4 || 4

6 || 6 || 6

8 || 8 || 8

10 || 10 || 10

12 || 12 || 12

14 || 14 || 14

16 || 16 || 16

18 || 18 || 18

20 || 20 || 20

22 || 22 || 22

24 || 24 || 24

26 || 26 || 26

28 || 42 || 42

30 || 50 || 50

32 || 58 || 58

34 || 66 || 66

36 || 91 || 91

38 || 105 || 105

40 || 119 || 119

42 || 161 || 161

44 || 187 || 187

46 || 213 || 213

48 || 239 || 239

50 || 333 || 333

52 || 383 || 383

54 || 433 || 433

56 || 595 || 595

58 || 693 || 693

60 || 791 || 791

62 || 889 || 889

64 || 1242 || 1242

66 || 1430 || 1430

68 || 1618 || 1618

70 || 2212 || 2212

72 || 2574 || 2574

74 || 2936 || 2936

76 || 3298 || 3298

78 || 4612 || 4612

80 || 5310 || 5310

82 || 6008 || 6008

84 || 8218 || 8218

86 || 9564 || 9564

88 || 10910 || 10910

90 || 12256 || 12256

92 || 17138 || 17138

94 || 19732 || 19732

96 || 22326 || 22326

98 || 30534 || 30534

100 || 35534 || 35534

Tiempos de ejecucion en segundos:

--------------------------------------------------------------------------------

n || Recursiva Cola || Iterativo

--------------------------------------------------------------------------------

0 || 0.00000 || 0.00000

2 || 0.00000 || 0.00000

4 || 0.00000 || 0.00000

6 || 0.00000 || 0.00000

8 || 0.00000 || 0.00000

10 || 0.00000 || 0.00000

12 || 0.00000 || 0.00000

14 || 0.00000 || 0.00000

16 || 0.00000 || 0.00000

18 || 0.00000 || 0.00000

20 || 0.00000 || 0.00000

22 || 0.00000 || 0.00000

24 || 0.00000 || 0.00000

26 || 0.00000 || 0.00000

28 || 0.00000 || 0.00000

30 || 0.00000 || 0.00000

32 || 0.00000 || 0.00000

34 || 0.00000 || 0.00000

36 || 0.00000 || 0.00000

38 || 0.00000 || 0.00000

40 || 0.00000 || 0.00000

42 || 0.00000 || 0.00000

44 || 0.00000 || 0.00000

46 || 0.00000 || 0.00100

48 || 0.00000 || 0.00000

50 || 0.00000 || 0.00000

52 || 0.00000 || 0.00000

54 || 0.00000 || 0.00000

56 || 0.00000 || 0.00000

58 || 0.00000 || 0.00000

60 || 0.00000 || 0.00000

62 || 0.00000 || 0.00000

64 || 0.00000 || 0.00000

66 || 0.00000 || 0.00000

68 || 0.00000 || 0.00000

70 || 0.00000 || 0.00000

72 || 0.00000 || 0.00000

74 || 0.00000 || 0.00000

76 || 0.00000 || 0.00000

78 || 0.00000 || 0.00000

80 || 0.00000 || 0.00000

82 || 0.00000 || 0.00000

84 || 0.00000 || 0.00000

86 || 0.00100 || 0.00000

88 || 0.00000 || 0.00000

90 || 0.00100 || 0.00000

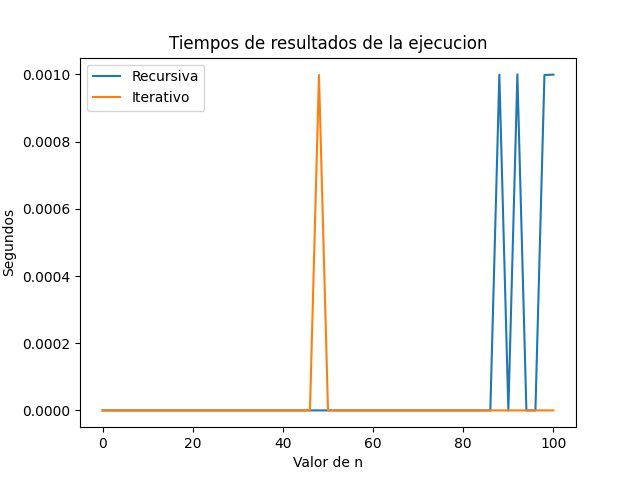
92 || 0.00000 || 0.00000

94 || 0.00000 || 0.00000

96 || 0.00100 || 0.00000

98 || 0.00000 || 0.00000

100 || 0.00300 || 0.00000



Comparacion de recursiva con la iterativa

Resultados de la ejecucion:

--------------------------------------------------------------------------------

n || Recursiva || Iterativo

--------------------------------------------------------------------------------

0 || 0 || 0

2 || 2 || 2

4 || 4 || 4

6 || 6 || 6

8 || 8 || 8

10 || 10 || 10

12 || 12 || 12

14 || 14 || 14

16 || 16 || 16

18 || 18 || 18

20 || 20 || 20

22 || 22 || 22

24 || 24 || 24

26 || 26 || 26

28 || 42 || 42

30 || 50 || 50

32 || 58 || 58

34 || 66 || 66

36 || 91 || 91

38 || 105 || 105

40 || 119 || 119

42 || 161 || 161

44 || 187 || 187

46 || 213 || 213

48 || 239 || 239

50 || 333 || 333

52 || 383 || 383

54 || 433 || 433

56 || 595 || 595

58 || 693 || 693

60 || 791 || 791

62 || 889 || 889

64 || 1242 || 1242

66 || 1430 || 1430

68 || 1618 || 1618

70 || 2212 || 2212

72 || 2574 || 2574

74 || 2936 || 2936

76 || 3298 || 3298

78 || 4612 || 4612

80 || 5310 || 5310

82 || 6008 || 6008

84 || 8218 || 8218

86 || 9564 || 9564

88 || 10910 || 10910

90 || 12256 || 12256

92 || 17138 || 17138

94 || 19732 || 19732

96 || 22326 || 22326

98 || 30534 || 30534

100 || 35534 || 35534

Tiempos de ejecucion en segundos:

--------------------------------------------------------------------------------

n || Recursiva || Iterativo

--------------------------------------------------------------------------------

0 || 0.00000 || 0.00000

2 || 0.00000 || 0.00000

4 || 0.00000 || 0.00000

6 || 0.00000 || 0.00000

8 || 0.00000 || 0.00000

10 || 0.00000 || 0.00000

12 || 0.00000 || 0.00000

14 || 0.00000 || 0.00000

16 || 0.00000 || 0.00000

18 || 0.00000 || 0.00000

20 || 0.00000 || 0.00000

22 || 0.00000 || 0.00000

24 || 0.00000 || 0.00000

26 || 0.00000 || 0.00000

28 || 0.00000 || 0.00000

30 || 0.00000 || 0.00000

32 || 0.00000 || 0.00000

34 || 0.00000 || 0.00000

36 || 0.00000 || 0.00000

38 || 0.00000 || 0.00000

40 || 0.00000 || 0.00000

42 || 0.00000 || 0.00000

44 || 0.00000 || 0.00000

46 || 0.00000 || 0.00000

48 || 0.00000 || 0.00100

50 || 0.00000 || 0.00000

52 || 0.00000 || 0.00000

54 || 0.00000 || 0.00000

56 || 0.00000 || 0.00000

58 || 0.00000 || 0.00000

60 || 0.00000 || 0.00000

62 || 0.00000 || 0.00000

64 || 0.00000 || 0.00000

66 || 0.00000 || 0.00000

68 || 0.00000 || 0.00000

70 || 0.00000 || 0.00000

72 || 0.00000 || 0.00000

74 || 0.00000 || 0.00000

76 || 0.00000 || 0.00000

78 || 0.00000 || 0.00000

80 || 0.00000 || 0.00000

82 || 0.00000 || 0.00000

84 || 0.00000 || 0.00000

86 || 0.00000 || 0.00000

88 || 0.00100 || 0.00000

90 || 0.00000 || 0.00000

92 || 0.00100 || 0.00000

94 || 0.00000 || 0.00000

96 || 0.00000 || 0.00000

98 || 0.00100 || 0.00000

100 || 0.00100 || 0.00000

Conclusiones

La implementación Recursiva es menos eficiente. Si se prueba con un valor mayor a 300 no habrá respuestas transcurridos 10 minutos. En los gráficos tambien se logra observar que en cuanto a resultados es menos eficiente en tiempos.

La implementación Recursiva de cola con un valor de n mayor que 1000 arroja el error: RecursionError: maximum recursion depth exceeded in comparison. Se debe a que se excedió el nivel de recursividad y por ende no se puede comprar con valores de n superiores a 1000.

Para n=100 se observa que la iterativa es más eficiente que la de cola pero por muy poco siendo casi iguales, pero para resultados con n mayores a 500 esto va aumentando

Finalmente se observa que la implementación Iterativa en la más eficiente en tiempo para n=1000 Y no posee un límite para un valor de n es decir puede intentar con cualquier valor.

codigo disponible en <https://github.com/JKOMI2303/Lenguaje_Programacion-1/tree/main/parcial%202/pregunta_4>

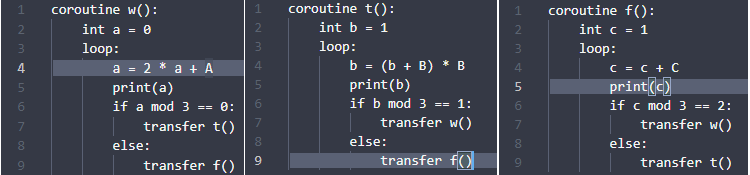
Pregunta 6:

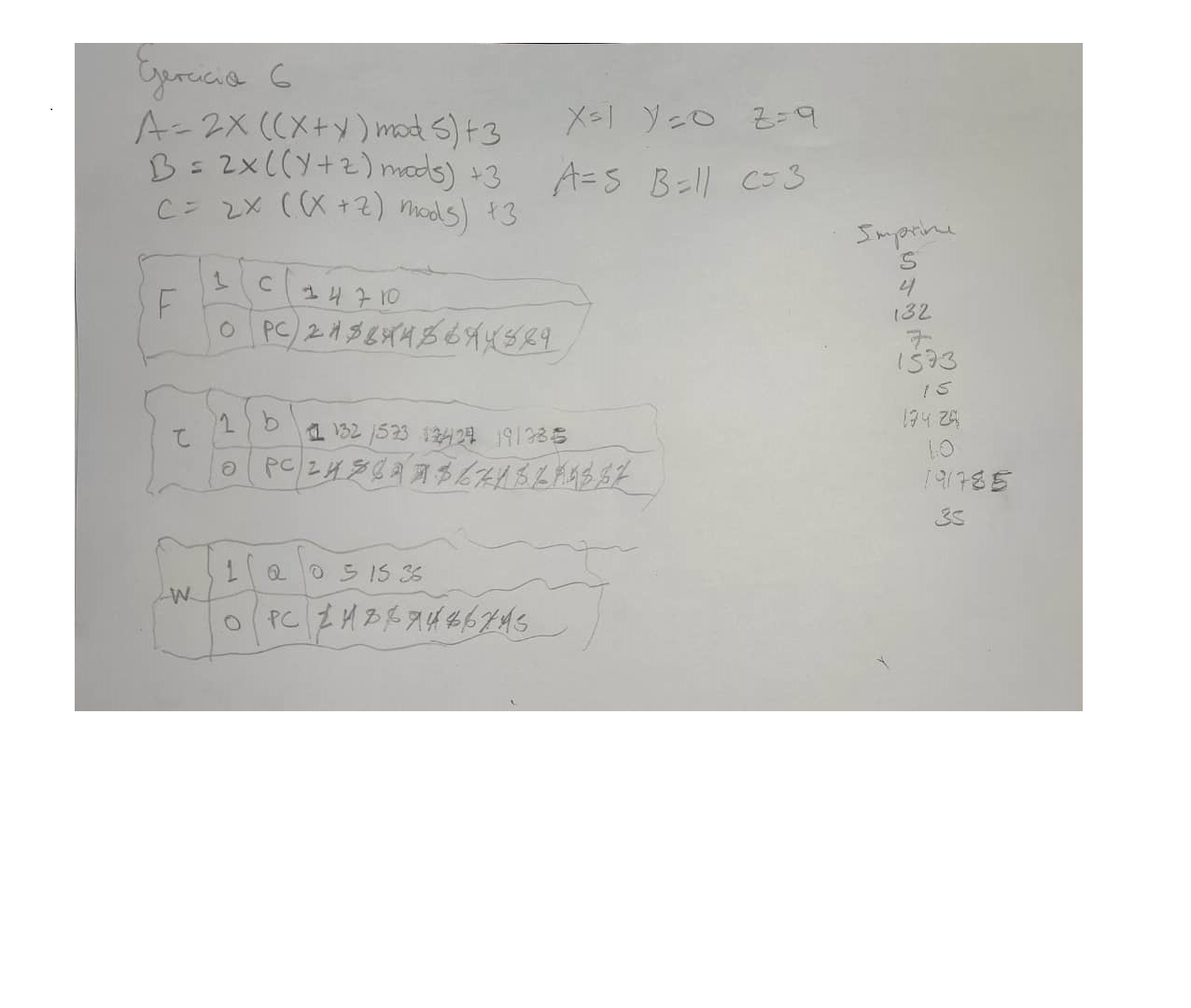
X=1 Y=0 Z=9

A=2x((X+Y)mod5)+3=5

B=2x((Y+Z)mod5)+3=11

C=2x((X+Z)mod5)+3=3





código de las corutinas disponibles en:

<https://github.com/JKOMI2303/Lenguaje_Programacion-1/tree/main/parcial%202/pregunta_6>